

Docket No.: 42390P13000
Express Mail No. EL651893697US

UNITED STATES PATENT APPLICATION

for

**AN APPARATUS AND METHOD FOR
WIRELESS/WIRED
COMMUNICATIONS INTERFACE**

Inventor:

Kelan C. Silvester

Prepared by:

Blakely, Sokoloff, Taylor & Zafman LLP
12400 Wilshire Boulevard
Seventh Floor
Los Angeles, CA 90025-1026
(310) 207-3800

AN APPARATUS AND METHOD FOR WIRELESS/WIRED COMMUNICATIONS INTERFACE

FIELD OF THE INVENTION

[0001] The invention relates generally to the field of wireless communications devices. More particularly, the invention relates to a method and apparatus for when a wireless/wired communications interface is detected as the communications configuration.

BACKGROUND OF THE INVENTION

[0002] The Bluetooth™ radio system is defined by a specification for a wireless, cable replacement solution based on radio frequency (RF) technology. In addition, Bluetooth™ has been specified and designed with emphasis on robustness and low cost. Implementations of Bluetooth™ are based on high performance, yet low-cost integrated radio transceivers. Accordingly, Bluetooth™ is defined as a protocol with a wireless interface. The frequency assignment that is utilized by Bluetooth™ within the United States is in the same radio frequency range as 802.11 wireless LAN (local area network), high end portable home phones, medical devices, as well as microwave ovens. As a result, the required nominal range of Bluetooth™ radio devices is set to approximately 10 meters, although distances may vary according to the power output of the radio.

[0001] Bluetooth™ is targeted at mobile and business users who need to establish a link or small network between their computers, cellular phones or other peripherals. To fill this communications need, a Bluetooth™ radio in a phone, PDA (personal digital assistant), headset, notebook computer or other device enables the devices to communicate with each other. As a result, Bluetooth™ is gaining in popularity as a wireless cable replacement technology. Moreover, Bluetooth™ provides a new market segment and devices to solve an age old problem of excessive number of wires used to connect personal computers (PCs) and other mobile devices. Unfortunately, devices in various environments cause radio traffic as well as interference problems in the radio frequencies utilized by Bluetooth™ devices.

[0004] As a result, communication information interference can become a problem when utilizing Bluetooth™-enabled devices. In addition, Bluetooth™ devices may cause interference problems with other devices. As described above, the frequency assignment to Bluetooth™ devices is in the same radio frequency range as 802.11 wireless LANs, high-end portable home phones, medical devices, as well as microwave ovens. Consequently, these other devices may cause interference with Bluetooth™-enabled devices. Conversely, Bluetooth™-enabled devices may affect the proper functioning of LANs, portable phones, medical devices, microwave ovens or the like.

[0005] Consequently, although Bluetooth™ devices may offer users the ability to replace the myriad of cables for attaching to various devices, the Bluetooth™ device is not immune to radio frequency interference. Hence, Bluetooth™ devices, when operated in the presence of significant radio frequency interference, will often perform poorly or even malfunction, which may lead to dissatisfaction of users of such Bluetooth™-enabled devices. This dissatisfaction may reach the point where users decline usage of Bluetooth™ devices and opt for standard wired connections, as are currently available in various devices described above. Therefore, there remains a need to overcome one or more of the limitations in the above-described existing.

BRIEF DESCRIPTION OF THE DRAWINGS

[0006] The present invention is illustrated by way of example, and not by way of limitation, in the figures of the accompanying drawings and in which:

[0001] FIG. 1 depicts a block diagram illustrating a Bluetooth™ piconet utilizing a host computer configured to implement a wireless/wired Bluetooth™ communications interface, in accordance with one embodiment of the present invention.

[0008] FIG. 2 depicts a block diagram further illustrating the host computer, as depicted in FIG. 1, in accordance with the further embodiment of the present invention.

[0009] FIG. 3 depicts a block diagram illustrating the Bluetooth™ controller, as illustrated in FIG. 2, in accordance with one embodiment of the present invention.

[00010] FIG. 4 depicts the host interface, as depicted in FIG. 3, in accordance with a further embodiment of the present invention.

[00011] FIG. 5 depicts a block diagram further illustrating the Bluetooth™ baseband controller, as depicted in FIG. 3, in accordance with the further embodiment of the present invention.

[00012] FIG. 6 depicts a Bluetooth™ software stack, implementing a Bluetooth™ baseband controlled driver to provide a wired/wireless Bluetooth™ communications interface, in accordance with an exemplary embodiment of the present invention.

[00013] FIG. 7 depicts a block diagram illustrating a Bluetooth™ software architecture utilizing the Bluetooth™ baseband control driver, as depicted in FIG. 6, and in accordance with a further embodiment of the present invention.

[00014] FIG. 8 depicts a flowchart illustrating a method for implementing a wired/wireless Bluetooth™ communications interface, in accordance with one embodiment of the present invention.

[00015] FIG. 9 depicts a flowchart illustrating an additional method, which is performed prior to detection of a device communications type, in accordance with a further embodiment of the present invention.

[00016] FIG. 10 depicts a flowchart illustrating an additional method for selecting a communications mode of the wireless device, in accordance with a further embodiment of the present invention.

[00017] FIG. 11 depicts a flowchart illustrating a method for switching a device communications mode from one of a wired communications mode to a wireless communications mode, in accordance with an exemplary embodiment of the present invention.

DETAILED DESCRIPTION

[0001] A method and apparatus for wireless communications devices are described. The method includes detection of a communications configuration of a device as one of a wireless communications interface and a wireless/wired communications interface. Once detected, it is determined whether a wired/wireless communications interface is the communications interface of the device. When such is the case, a data transmission error rate of the device is determined during operation of the device in a wireless communications mode. Next, the device is switched from the wireless communications mode to a wired communications mode when the data transmission error rate of the device exceeds a pre-determined threshold. As such, the present invention allows Bluetooth™ wireless communications devices to utilize a wired communications mode when transmission error rate exceeds a pre-determined threshold.

[00019] In the following description, for the purposes of explanation, numerous specific details are set forth in order to provide a thorough understanding of the present invention. It will be apparent, however, to one skilled in the art that the present invention may be practiced without some of these specific details. In addition, the following description provides examples, and the accompanying drawings show various examples for the purposes of illustration. However, these examples should not be construed in a limiting sense as they are merely intended to provide examples of the present invention rather than to provide an exhaustive list of all possible implementations of the present invention. In other instances, well-known structures and devices are shown in block diagram form in order to avoid obscuring the details of the present invention.

[00020] Portions of the following detailed description may be presented in terms of algorithms and symbolic representations of operations on data bits. These algorithmic descriptions and representations are used by those skilled in the data processing arts to convey the substance of their work to others skilled in the art. An algorithm, as described herein, refers to a self-consistent sequence of acts leading to a desired result. The acts are those requiring physical manipulations of physical quantities. These quantities may take the form of electrical or magnetic signals capable of being stored, transferred, combined, compared, and otherwise manipulated. Moreover, principally for reasons of common usage, these signals are referred to as bits, values, elements, symbols, characters, terms, numbers, or the like.

[00021] However, these and similar terms are to be associated with the appropriate physical quantities and are merely convenient labels applied to these quantities. Unless specifically stated otherwise, it is appreciated that discussions utilizing terms such as “processing” or “computing” or “calculating” or “determining” or “displaying” or the like, refer to the action and processes of a computer system, or similar electronic computing device, that manipulates and transforms data represented as physical (electronic) quantities

within the computer system's devices into other data similarly represented as physical quantities within the computer system devices such as memories, registers or other such information storage, transmission, display devices, or the like.

[00022] The algorithms and displays presented herein are not inherently related to any particular computer or other apparatus. Various general purpose systems may be used with programs in accordance with the teachings herein, or it may prove convenient to construct more specialized apparatus to perform the required method. For example, any of the methods according to the present invention can be implemented in hard-wired circuitry, by programming a general-purpose processor, or by any combination of hardware and software.

[00023] One of skill in the art will immediately appreciate that the invention can be practiced with computer system configurations other than those described below, including hand-held devices, multiprocessor systems, microprocessor-based or programmable consumer electronics, digital signal processing (DSP) devices, network PCs, minicomputers, mainframe computers, and the like. The invention can also be practiced in distributed computing environments where tasks are performed by remote processing devices that are linked through a communications network. The required structure for a variety of these systems will appear from the description below.

[00024] It is to be understood that various terms and techniques are used by those knowledgeable in the art to describe communications, protocols, applications, implementations, mechanisms, etc. One such technique is the description of an implementation of a technique in terms of an algorithm or mathematical expression. That is, while the technique may be, for example, implemented as executing code on a computer, the expression of that technique may be more aptly and succinctly conveyed and communicated as a formula, algorithm, or mathematical expression.

[00025] Thus, one skilled in the art would recognize a block denoting " $C=A+B$ " as an additive function whose implementation in hardware and/or software would take two inputs (A and B) and produce a summation output (C). Thus, the use of formula, algorithm, or mathematical expression as descriptions is to be understood as having a physical embodiment in at least hardware and/or software (such as a computer system in which the techniques of the present invention may be practiced as well as implemented as an embodiment).

[00026] In an embodiment, the methods of the present invention are embodied in machine-executable instructions. The instructions can be used to cause a general-purpose or special-purpose processor that is programmed with the instructions to perform the steps of the present invention. Alternatively, the steps of the present invention might be performed by specific hardware components that contain hardwired logic for performing

the steps, or by any combination of programmed computer components and custom hardware components.

[00027] In one embodiment, the present invention may be provided as a computer program product which may include a machine or computer-readable medium having stored thereon instructions which may be used to program a computer (or other electronic devices) to perform a process according to the present invention. Accordingly, the computer-readable medium includes any type of media/machine-readable medium suitable for storing electronic instructions. Moreover, the present invention may also be downloaded as a computer program product. As such, the program may be transferred from a remote computer (e.g., a server) to a requesting computer (e.g., a client). The transfer of the program may be by way of data signals embodied in a carrier wave or other propagation medium via a communication link (e.g., a modem, network connection or the like).

System Architecture

[0001] Referring now to FIG. 1, FIG. 1 depicts a Bluetooth™ piconet 100, including a master, host computer 200, which implements a wired/wireless Bluetooth™ communications interface. As known to those skilled in the art, a piconet describes a collection of devices connected via Bluetooth™ technology in an ad hoc fashion. A piconet is initialized with two connective devices, such as for example, a portable personal computer and a mobile phone. The number of devices which can participate in a piconet is limited to eight units. Accordingly, as illustrated in FIG. 1, the various devices, including headset 110, joystick 120, mouse keyboard 130, personal digital assistant (PDA) 140, laptop computer 150, fax machine 160 and printer 170, may participate in a piconet with host computer 200.

[0001] As described above, piconets, such as piconet 100 as depicted in FIG. 1, are designed as wire replacement technologies for implementing such networks as depicted in FIG. 1 without the need of wires for interconnecting devices. As such, a host computer would merely await page requests from the various devices in order to establish a connection between the host computer 200 and an authorized device, such as for example, headset 110. Unfortunately, Bluetooth™ enabled devices, such as for example, headset 110, are subject to various interference problems. As interference occurs, the host device may lose its connection with the Bluetooth™ device. Furthermore, interference problems may result in lost or dropped data packets from the Bluetooth™ connection between devices. In the case of a Bluetooth™ audio headset connected to a portable computer, the lost or dropped data packets would result in crackle or pops or gaps in the audio stream received at the headset.

[00030] Accordingly, as interference levels are detected, they can eventually reach the limit of intolerable interference within the Bluetooth™ devices, such that even the frequency hopping and error correction algorithms enabled by Bluetooth™ devices is not

sufficient to overcome the interference problems. As a result, the present invention describes Bluetooth™ devices which implements a baseband controller, which enables a wired/wireless Bluetooth™ communications interface. Consequently, the various Bluetooth™ devices may either communication in a wireless fashion with a host device or be connected with the host device via a wired link.

[00031] As such, the wire link may be used whenever desired and is, in fact, required when a transmission data error rate between the host device and the various Bluetooth™ devices exceeds a pre-determined level. For example, in the case of a Bluetooth™ device transmitting audio data, an error rate of <1% may be established as the threshold error rate. Error rates in data traffic result in slowed performance of the Bluetooth™ link as it retransmits data until errors are corrected. In one embodiment, performance drops of >20% may necessitate switching from a wireless link to a wired link.

[00032] However, the various baseband controllers within the devices utilize the same communication methodology as conventional Bluetooth™ devices. Nevertheless, communication of voice and data is now provided between the devices using a wired link. In certain embodiments, the Bluetooth™ device may encounter excessive network interference requiring switching to a wired communications mode. When a wire is not currently being utilized, the devices may either direct themselves or require the host device to notify a user to connect a wire link between the devices to enable the wired communications mode. However, once the communication interference has subsided, the user can be notified to remove the wired link, if desired, and return to wireless (radio frequency (RF)) communication between the devices. In one embodiment, the configuration change may be done automatically by the system according to predefined preference settings.

[0001] Referring now to FIG. 2, FIG. 2 further illustrates the host computer 200, as depicted in FIG. 1, in accordance with a further embodiment of the present invention. As depicted, the host computer 200 includes a CPU 302, a user interface 204, a communications interface 206, as well as a memory 212. However, the host computer 200 is implemented utilizing a Bluetooth™ controller 300, which includes a wire link connection port 302 in order to implement a wired/wireless interface, either via wire link port 302 or Bluetooth™ radio 310.

[00034] Although the host computer 200 is illustrated using the Bluetooth™ controller 300, those skilled in the art will recognize that each of the Bluetooth™ enabled devices, such as monitor 260, keyboard 262 and mouse 264 also include a Bluetooth™ control and Bluetooth™ radio as depicted in FIG. 2. Accordingly, the Bluetooth™ enabled monitor 260, keyboard 262 and mouse 264 can be utilized by the host control in order to provide a user interface without the need for interconnected wires between the host

computer 200 and the Bluetooth™ devices. However, for the reasons described above, data transmission errors may be incurred during times of high interference levels.

[00035] Consequently, when interference levels exceed a pre-determined level, it may be necessary for a user to interconnect the various Bluetooth™ devices with a wired link to the wired link port 302 of the Bluetooth™ controller 300. The various devices may themselves determine when excessive interference levels are detected and notify the user to make the wired connection. Alternatively, the Bluetooth™ controller 300 may make such determinations and notify the user to interconnect the wire link devices. In other embodiments, the detection of excessive error conditions resulting in a switch from wireless to wired transmission can be made automatically by either the device or the Bluetooth™ controller. As a result, the Bluetooth™ enabled devices will switch to a wired communications mode once notified by the Bluetooth™ controller 300 of a host device 200.

[00036] Referring again to FIG. 2, the memory 212 of the host computer 200 may be implemented as RAM (random access memory), SRAM (synchronous random access memory), SDRAM (synchronous data random access memory) or a combination of RAM and non-volatile memory, such as one or more memory modules, storage volumes or magnetic disc storage units. The memory can contain any of the following:

- an operating system 214, which includes for example, windows driver model (WDM drivers) such as the human interface device (HID) class driver, streaming media windows drivers model, as well as the network driver interface specification supported by the operating system, such as for example, Winsock, transmission communication protocol (TCP), Internet protocol (IP) (TCP/IP);
- Bluetooth™ driver interface 216;
- Bluetooth™ host controller driver 220;
- base band control driver 222;
- communications interface type detection procedures 224;
- communications mode detection procedures 226;
- communications mode switch procedures 228;
- wire link detection procedures 230;
- error rate detection procedures 232;
- user notification procedures 234;
- RF.com driver 236;
- human interface device (HID) mini-drivers 238;

- audio mini-drivers 240;
- network mini-drivers 242; and
- other procedures and files.

[00037] Accordingly, the various procedures and drivers included in the host computer memory 212 will be described in further detail below within the Software Architecture description.

[0001] Referring now to FIG. 3, FIG. 3 further illustrates the Bluetooth™ controller 300, as depicted in FIG. 2, in accordance with a further embodiment of the present invention. As depicted, the Bluetooth™ controller 300 includes an analog portion as the Bluetooth™ radio 310, which is utilized to communicate with the various Bluetooth™ enabled devices, as well as the digital portion, the link controller (Bluetooth™ baseband controller) 500. Accordingly, the Bluetooth™ baseband controller 500 performs the digital signal processing functions of the Bluetooth™ controller using various signal processing hardware. The Bluetooth™ controller 300 further includes a CPU (central processing unit) core 360, which is optional, but is however beneficial, in order to avoid interface problems with a personal computer CPU. The CPU core 402 is further utilized to interface between external interfaces 404, which provide a host interface 400 to the various additional devices of the host computer 200.

[00039] As such, the baseband controller 500 includes hardware for performing baseband processing and basic protocols close to the physical layer, such as for example, ARQ (automatic request repeat) protocol and FEC (forward error control) coding. Hence, the baseband controller 500 incorporates all hardware required to interface the Bluetooth™ controller 300 to the Bluetooth™ radio environment. Generally, command responses and data are transferred between Bluetooth™ units across the radio interface in packet format. In addition, the baseband controller 500 is generally responsible for quality of service parameters, asynchronous transfers with guaranteed delivery, synchronous transfers audio coding and encryption.

[00040] Referring again to FIG. 3, the link manager 350 is a software entity which carries out protocols, such as link set-up, authentication link configuration and the like. The link manager discovers other remote link managers and communicates with them via the link management protocol to perform service provider role and uses the services of the underlying baseband controller. The service provided includes name requests, which may be up to 16 characters in length, link address inquiries, connection set-up, authentication, link mode negotiation and set-up, such as for example, data or data invoice, setting devices in sniff mode, hold mode, park mode and active mode.

[00041] Referring now to FIG. 4, FIG. 4 depicts host interface 400, as depicted in FIG. 3, in accordance with one embodiment of the present invention. The host interface

(HIF) 400 includes hardware and software, which interfaces the Bluetooth™ controller 300 to a host, such as for example, host device 200, as depicted in FIG. 2. The primary function of HIF 400 is to provide services of the lower layers (via the link manager) in a format suitable for the host device.

[00042] Accordingly, the various external interfaces are depicted in FIG. 4, which may include for example, PC card interface 430, PCI (peripheral component interconnect) interface 440, low pin out pin count interface 450, USB interface 460 and dual port memory interface 470. These devices may implement and support PC card bus RS 232, PC mother board integration, as well as USB slave 462. As such, the Bluetooth™ controller firmware implements the baseband link management protocol. The drivers control the radio 310 using the Bluetooth™ host controller interface 400, which is accessed through an appropriate transport interface, as will be described in further detail below.

[00043] Referring now to FIG. 5, FIG. 5 further illustrates the Bluetooth™ baseband controller, as depicted in FIG. 4. The Bluetooth™ baseband controller includes a processor 402 along with memory arbiter 540, SRAM sequencer 550, flash memory 560 and SRAM 570. However, in contrast to conventional baseband controllers, the Bluetooth™ baseband controller 500 includes Bluetooth™ baseband 520, which utilizes radio/wire detection block 510 in order to determine a communications mode of the respective device and to determine whether a wire link between a host device and the Bluetooth™ device is connected. As such, the Bluetooth™ baseband 520 includes firmware for performing the radio/wire detection functionality in order to perform the baseband control driver procedures 222-230, as depicted in FIG. 2.

[00044] Accordingly, wire connection block 530 is utilized by the Bluetooth™ baseband controller 500 in order to provide a physical wire link to a slave device. Therefore, when interference levels exceeds a pre-determined level, a host device can notify a user to connect a wire link between the device and a host device via wire connection block 530. Consequently, using wire link Bluetooth™ devices can avoid unduly interference levels and communicate via the wire link until the interference levels have subsided. Once interference levels have subsided, the various devices can notify the user to remove the wire link and resume wireless communication between the various devices. Alternatively, switching between modes is performed automatically according to the detected interference levels.

Software Architecture

[0001] The software architecture for implementing Bluetooth™ devices is based on the concept of treating the short range radio, such as for example, Bluetooth™ RF 310, as depicted in FIGS. 2-4, as a bus driver. Treating the local radio 310 as a bus controller enables the loading of appropriate device drivers on detection and identification of other Bluetooth™ compliant devices within range. As such, Bluetooth™ devices fall under the

following three classes – telephone related devices, traditional PC peripherals and communication/network devices.

[0001] In order to support Bluetooth™ peripherals, the software model 600 utilizes the windows driver model (WDMS) human interface (HID) class driver. The HID class driver comes standard from, for example, Microsoft's NT™ 5.0 and 98 operating systems, and provides support for all human interface devices, such as bus list mechanisms. The HID driver is capable of identifying the HID peripheral and loading the corresponding HID mini drivers in a way that is bus independent. The HID class driver must be pointed to a WDM bus driver. Accordingly, by means of the Bluetooth™ WDM bus driver and an HID to Bluetooth™ bus mini driver, the HID class driver can be used to support all human interface devices, such as mice, keyboard, joysticks or the like.

[0001] Support for real time voice has been implemented using digital audio hooks in the PC via the streaming media Windows driver model (WDM) defined by Microsoft Corporation of Redmond, Washington. In addition, the network driver interface specification has connection orientation channels at the (NDIS) layer, which targets telephone applications. Finally, in networking applications are implemented by utilizing the Bluetooth™ controller interface as a network interface card (NIC), utilizing a mini driver to interface the NDIS class driver with the Bluetooth™ WDM bus driver.

[00048] Referring now to FIG. 6, FIG. 6 depicts the Bluetooth™ software stack 600. As illustrated, the core of the Bluetooth™ software stack 600 essentially includes the Bluetooth™ bus driver 630, the Bluetooth™ host control interface (HCI) 640 and the Bluetooth™ host controller driver 650. Accordingly, each function class typically includes a client driver that is loaded by the Bluetooth™ bus driver. These client drivers utilize the Bluetooth™ bus driver interface 620 to communicate with the Bluetooth™ bus driver 630 for data and control transfer purposes.

[00049] However, in contrast to conventional Bluetooth™ software stacks, Bluetooth™ software stack 600 includes a Bluetooth™ or link manager control driver 660, which is utilized to implement the wired/wireless communications interface, as described herein. However, those skilled in the art will recognize that the wired/wireless interface communications methods described herein may be implemented anywhere within the Bluetooth™ bus driver interface 620 software stack 600, as well as within the Bluetooth™ controller firmware, depending on the desired implementation specific details of the system designer.

[00050] As such, a Bluetooth™ software stack 600, as depicted in FIG. 6, is further illustrated in the Bluetooth™ software architecture 700, as depicted in FIG. 7. As illustrated, user mode 710 programs interact with the kernel 720 using a variety of programmer interfaces, including, for example, WIN32™, Winsock™, RealTime™ applications server (RAS) and telephone application program interface (TAPI). As such,

Bluetooth™ kernel mode drivers 720 fit into the framework exposed by Windows 98™ and Windows 2000™ for third party drivers. Although the Bluetooth™ software architecture 700, as depicted in FIG. 7, is illustrated with reference to application programming interfaces specific to Windows 98™ and Windows 2000™, those skilled in the art will recognize that various third party drivers available in computing operating systems, such as for example, Unix, Linux or the like, may be provided in order to provide and implement Bluetooth™ device specific interfaces.

[0001] Bluetooth™ kernel mode drivers 720 fall into two categories: one set of drivers provide the infrastructure on which the second set of client Bluetooth™ drivers are layers. Infrastructure drivers are the ones below the Bluetooth™ driver interface. Bluetooth™ client drivers interface to the main specific interfaces in the operating system. As illustrated, FIG. 7 depicts the software architecture for support, including networking communications, TAPI, HID, still image and audio. However, those skill in the art will recognize that this is not an exhaustive list. In addition, the Bluetooth™ host controller implements a baseband controlled driver, as illustrated in FIG. 6.

[00052] Accordingly, as illustrated in FIG. 2, within the memory of the host device 200, the baseband control driver procedures 222 enable the wireless/wired communications interface, as referred to above, within Bluetooth™ enabled devices. As such, the communication interface type detection procedures 224 are utilized by the baseband control driver procedures 222 in order to determine whether the slave device is configured with a wired/wireless communications interface. When such is the case, the procedures can use the communications mode detection procedures 226 to determine whether a wire link is being used for communication between the devices.

[00053] As such, when a wireless mode is detected, the error rate detection procedures 232 will monitor packet error rates to determine interference levels. When the interference levels exceed a pre-determined level, the user notification procedures 234 will be utilized to notify a user to connect a wire link between the devices. Once a connection is made, the baseband control driver procedures 222 will invoke the communicator mode switch procedures 228 to switch communication from the wireless mode to the wired mode via wire link 320.

[0001] In addition, the host device 200 includes a variety of mini drivers (232-240) for implementing the Bluetooth™ specific devices. In addition, the user mode applications include a Bluetooth™ advisor, as provided as a single point of control for all Bluetooth™ devices. It is generally targeted to be the end user visible application for Bluetooth™ related configurations. Within the infrastructure drivers described, the USB mini driver 730, is a client of Microsoft's USB stack and communicates using the USB driver interface (USBDI). Accordingly, the driver provides a transport mechanism for access to the Bluetooth™ host controller interface 640. The Bluetooth™ HCI 640 abstracts driver

abstracts the HIC interface from the Bluetooth™ bus driver 630 and publishes a much simplified connection model. The Bluetooth™ bus driver 630 is one of the core components of the Bluetooth™ stack. For Bluetooth™ purposes, the RF interface is modeled as a peripheral bus on which Bluetooth™ devices come and go.

[0001] The Bluetooth™ bus driver 630 is the bus driver for the RF bus and is responsible for the following functions: device enumeration, connection maintenance, client driver loading, implementation, segmentation, reassembly and protocol multiplexing, support for RF bus driver interface (RFB DI) management of radio transmissions and management of power states in the Bluetooth™ device. RF communication driver is a client of the Bluetooth™ bus and is responsible for implementing comport import immolation over Bluetooth™. It is generally used for dial-up networking and speech applications. It can also be used as a transport for objects exchange (OBEX) insert and implementations. OBEX enables synchronization applications between various Bluetooth™ devices, such as between a notebook computer and a phone.

[00056] The HID mini drivers 722 provide support for HID via the phone through an HID mini driver that interfaces with the HID class driver and the Bluetooth™ bus driver 630. These interfaces are also utilized in order to enable support for other HID devices in Bluetooth™, including keyboards, joysticks, mice and game pads. An audio mini driver is also provided that interfaces with the streaming class driver from the Microsoft operating system. It is utilized as a capacitor from the system audio devices to the Bluetooth™ interface, where it will be transported to a headset, cellular phone or another notebook computer. Network mini drivers also are provided in order to provide LAN access points, as well as conference table functions. Accordingly, the procedural methods for implementing the wired/wireless communications interface as described herein are now provided.

Operation

[0001] Referring now to FIG. 8, FIG. 8 depicts a method 800 for implementing a wired/wireless Bluetooth™ communications interface within, for example, the Bluetooth™ devices as depicted in FIGS. 1 and 2. As described above, the various wired/wireless interface procedures, as described herein, may be implemented utilizing the baseband control driver 222, as depicted in FIGS. 2, 6 and 7. Alternatively, the wired/wireless interface procedures may be implemented utilizing software procedures within the link manager 350, as depicted in FIG. 3. Otherwise, the interface procedures may be implemented utilizing baseband controller firmware within baseband controller 500, as depicted in FIG. 5.

[0001] Accordingly, utilizing the wired/wireless interface techniques describes herein, Bluetooth™ enabled devices may utilize a wire link to communicate with a Bluetooth™ host device when environment interference exceeds a pre-determined level,

which prohibits wireless communication between the various devices. However, as described herein, communication continues via the Bluetooth™ baseband controller, with the exception that the Bluetooth™ radio is no longer utilized as the various information packets of data invoice information are transmitted through a wire link that connects the various devices.

[0001] Referring again to FIG. 8, at process block 840, a communications interface type of a device is detected. In the embodiments described, the communications interface type is one of a wireless communications interface and a wired/wireless communications interface. As described herein, the wireless communications interface refers to conventional Bluetooth™ baseband controllers, which are limited to a wireless interface. However, Bluetooth™ enabled devices utilizing baseband controllers, as described herein, will implement a wireless/wired communications interface.

[00060] Accordingly, at process block 842, when a wired/wireless communications interface is detected, at process block 844, a communications mode of the device is detected. In one embodiment, the interface type and the communications mode of the device are determined by querying a communications interface type state field and a wired link state field, which are set during, for example, start-up of the device. As described herein, the communications mode of the device includes wireless communications mode and a wired communications mode for dual wired/wireless communications interfaces.

[00061] Next, at process block 846, it is determined whether the device is functioning in a wireless communications mode. When such is the case, at process block 848, a data transmission error rate of the device is determined. Once determined, at process block 850, it is determined whether the data transmission error rate exceeds a pre-determined threshold error rate. As such, process block 848 is repeated until the error rate exceeds a pre-determined threshold. Accordingly, at process block 852, when the data transmission error rate exceeds the pre-determined threshold error rate, the wireless communications mode of the device is switched from the wireless communications mode to a wired communications mode. Alternatively, an environment interference level may be determined in order to decide whether to initiate communication in a wired communications mode.

[00062] As such, the device will now communicate using the baseband controller via a wired link interconnecting device with a host device. As indicated, the various procedural methods may either be implemented solely within the Bluetooth™ device or within a host device, such that a host device could send a broadcast packet to the various Bluetooth™ devices to begin functioning in the wired communications mode. In addition, when a wire link is not present, the host device could notify a user to plug in the required wired link.

[0001] Alternatively, such notification could be provided via the devices, themselves, such that the host device begins functioning in a wired communications mode

upon notification from a device or when plugin of the wired link is detected. The various baseband controlled driver procedures are described with reference to FIG. 2 and include communications interface procedures 224, communications mode detection procedures 226, communications mode switch procedures 228, wire link detection procedures 230, error rate (interference level) detection procedures 232 and user notification procedures 234, which may be implemented solely in the host computers as well as in the various Bluetooth™ enabled devices utilizing the Bluetooth™ software stack or baseband controller firmware.

[00064] Referring now to FIG. 9, FIG. 9 illustrates a method 802 which is performed prior to detection of the device communications interface type. Accordingly, at process block 804, a device communications interface type is determined. Once the interface type of the device is determined, a device communications interface state is set according to the determined communications interface type. Next, at process block 808, it is determined whether a wired/wireless communications interface is detected as the communications configuration of the device. When such is the case, at process block 810, it is determined whether a wired link couples the device to a host device. When such is the case, at process block 812, a wired connection state is set. Next, at process block 814, a communications mode of the device is selected according to the device communications configuration state, the wired connections state and a radio frequency interference level.

[00065] Finally, at process block 816, a device communications mode state is set according to the selected communications mode. As described above, the communications mode will be selected as a wireless communications mode when interference levels are at a normal level. However, when interference levels exceed a pre-determined threshold error level, the device will switch to a wired communications mode assuming a dual wired/wireless interface is supported. As will be described in detail below, when a wire link is not present during times of excessive interference, a user of a dual wired/wireless interface device will be notified to connect the device to a host device via a wire link.

[00066] Referring now to FIG. 10, FIG. 10 depicts a flowchart illustrating an additional method 816 for selecting the communications mode of process block 816, as depicted in FIG. 10. At process block 822, it is determined whether a wireless/wired communications interface of the device is detected. When such is the case, at process block 824, it is determined whether a wire link couples the device to a host device. When such is the case, at process block 826, a radio frequency interference level is determined. Once determined, at process block 828, it is determined whether the radio frequency interference level exceeds a pre-determined interference level. As such, process block 826 is repeated until the radio frequency interference level exceeds the pre-determined interference level. As such, once this case is determined, the communications mode of the device is set to the wired communications mode at process block 830.

[00067] In one embodiment, process blocks 826 and 828 are optional. In this embodiment, when a wire link is detected, the device will continually function in the wired communications mode. However, in the embodiment described with reference to FIG. 10, the device can automatically switch between communication modes (wired/wireless), as dictated by the interference levels, without notifying the user. Alternatively, the user may be notified to disconnect the wire link and enjoy wireless freedom provided by the wireless communications mode.

[00068] Referring now to FIG. 11, FIG. 11 depicts a flowchart illustrating a method for switching the device communications mode. At process block 862, the determined data transmission error rate is compared to a pre-determined threshold error rate. Next, at process block 824, it is determined whether the data transmission error rate exceeds the pre-determined threshold error rate. When such is the case, the device communications mode is set to the wired communications mode. Once set to the wired communications mode, at process block 868, communication of the device is initiated via a wire link between the device and a host device. Otherwise, at process block 870, the device communications mode is set to a wireless communications mode. Finally, at process block 880, communication of the device is initiated via a wireless link, such as for example, Bluetooth™ radio 310.

[00069] In alternate embodiments, during communication of the device according to the wired communications mode, one of the device or the host device will continue monitoring of the device interference levels or transmission error rates in order to determine when interference levels leading to increased transmission error rates have subsided. As such, once interference levels and corresponding data transmission error rates have fallen within an accepted threshold level, a user may be notified to disconnect the wire link between the device and the host device. Consequently, once the wire link is removed, the devices can resume functioning in a wireless communications mode, thereby enabling a user of the device to move around freely without being constricted by wires. As a result, this process may be continually repeated as interference levels are encountered by the user.

[00070] In a further alternate embodiment, a wire link is continuously connected between the wireless device and the host device. In such an embodiment, one of the host device or the wire device determines the interference levels within the environment and selects the communications mode accordingly. As indicated above, during times of excess interference levels, either of the devices will switch to the wired communications mode. However, when the interference level returns to an acceptable range, the devices may return to a wireless communications mode automatically, without alerting the user. This embodiment is provided in order to enable the user to have continued, uninterrupted communication.

[00071] Nonetheless, the user is restricted to having a wire link in place at all times during communications between the devices. However, the user can, at their option, remove the wire links and function in a wireless communications mode until an excessive interference level is detected, at which time the user may be requested to connect the wire link. As recognized by those skilled in the art, the various implementation details are up to the respective system designers and will include considerations relative to user preference for eliminating wire connections, as well as considerations for continuous communications and uninterrupted communication between the various devices.

[00072] As such, the process described herein will be repeated, requiring connecting a wire link and removing the wire link as necessary, in order to provide continued communication according to either a wired communications mode or a wireless communications mode. Moreover, voice/data communication between devices simply is performed by transmitting voice/data via either a wire link or a radio frequency transmitter within devices without requiring modification to device communication protocols or packet processing. Therefore, a user may continue communication without being subject to inoperable wireless devices during times of excessive interference levels leading to untenable error transmission rates.

Alternate Embodiments

[00073] Several aspects of one implementation of the when a wireless/wired communications interface is detected as the communications configuration for providing wired communication in the event of RF interference have been described. However, various implementations of the when a wireless/wired communications interface is detected as the communications configuration provide numerous features including, complementing, supplementing, and/or replacing the features described above. Features can be implemented as part of the wireless device or as part of a computer system in different implementations. In addition, the foregoing description, for purposes of explanation, used specific nomenclature to provide a thorough understanding of the invention. However, it will be apparent to one skilled in the art that the specific details are not required in order to practice the invention.

[00074] In addition, although an embodiment described herein is directed to a Bluetooth™ device, it will be appreciated by those skilled in the art that the teaching of the present invention can be applied to other systems. In fact, systems for limited distance wireless communication are within the teachings of the present invention, without departing from the scope and spirit of the present invention. The embodiments described above were chosen and described in order to best explain the principles of the invention and its practical applications. These embodiment were chosen to thereby enable others skilled in the art to best utilize the invention and various embodiments with various modifications as are suited to the particular use contemplated.

[00075] It is to be understood that even though numerous characteristics and advantages of various embodiments of the present invention have been set forth in the foregoing description, together with details of the structure and function of various embodiments of the invention, this disclosure is illustrative only. In some cases, certain subassemblies are only described in detail with one such embodiment. Nevertheless, it is recognized and intended that such subassemblies may be used in other embodiments of the invention. Changes may be made in detail, especially matters of structure and management of parts within the principles of the present invention to the full extent indicated by the broad general meaning of the terms in which the appended claims are expressed.

[00076] The present invention provides many advantages over known techniques. The present invention includes the ability to provide a wired interface to a Bluetooth™ device without the necessity to replace the protocol and foundation of the communications device. For example, a Bluetooth™-enabled keyboard can be used in a standard wireless fashion or it can be built such that it connects to a PC via a cable to perhaps a Bluetooth™ to USB connection. In this example, the user can chose either the wireless mode or can also plug the same keyboard into a USB port in cases where radio traffic is a problem. Accordingly, the present invention provides a back-up method of connecting devices where radio frequency traffic is a problem. In addition, devices are provided with a wired or wireless option. There are many circumstances when a user may not want to (or not be able to) use a wireless device, such as for example, battery depletion or the like. Finally, the present invention provides a mechanism for redundant connection or useful so that connectivity is not lost and protocol device connection does not have to ever change.

[00077] Having disclosed exemplary embodiments and the best mode, modifications and variations may be made to the disclosed embodiments while remaining within the scope of the invention as defined by the following claims.